



University  
of Glasgow

Rashidi, L., Rajasegarar, S., Leckie, C., Nati, M., Gluhak, A., Imran, M. A., and Palaniswami, M. (2014) Profiling Spatial and Temporal Behaviour in Sensor Networks: a Case Study in Energy Monitoring. In: 2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), Singapore, 21-24 Apr 2014, ISBN 9781479928439.

There may be differences between this version and the published version. You are advised to consult the publisher's version if you wish to cite from it.

<http://eprints.gla.ac.uk/136578/>

Deposited on: 9 February 2017

Enlighten – Research publications by members of the University of Glasgow  
<http://eprints.gla.ac.uk>

# Profiling Spatial and Temporal Behaviour in Sensor Networks: A Case Study in Energy Monitoring

Lida Rashidi\*, Sutharshan Rajasegarar<sup>†</sup>, Christopher Leckie\*,

Michele Nati<sup>‡</sup>, Alexander Gluhak<sup>‡</sup>, Muhammad Ali Imran<sup>‡</sup> and Marimuthu Palaniswami<sup>†</sup>

\*NICTA Victoria Research Laboratory, Dept. of Computing and Information Systems, <sup>†</sup>Dept. of Electrical and Electronic Eng.

<sup>‡</sup>The University of Melbourne, Australia; <sup>‡</sup>CCSR, University of Surrey, UK

{lrashidi@student., sraja@, caleckie@, palani@}unimelb.edu.au; {m.nati, a.gluhak, m.imran}@surrey.ac.uk

**Abstract**—Wireless sensor networks (WSNs) provide a cost-effective platform for monitoring phenomena of interest at fine spatial and temporal resolutions. In this paper, we consider the application of monitoring power usage in an office environment at the resolution of individual users. A key challenge in this context is how to extract meaningful profiles of user behaviour in the large volume of monitoring data collected by the WSN. To manage the complexity of learning such profiles in this context, we propose a query based model for profiling. This query based model provides the ability to characterize the spatial and temporal occurrences of the power usage patterns of interest. We demonstrate the effectiveness of our query-based profiling model for finding relevant electricity usage patterns in a real life data set of power measurements collected by a WSN deployment in an office environment. To the best of our knowledge, this is the first time such a case study has been made on analysing the power usage of users at such a fine scale in an office environment.

## I. INTRODUCTION

Wireless sensor networks (WSNs) provide the ability to monitor environments at high spatial and temporal resolutions. In this paper, we focus on the application of using a WSN to monitor electrical power usage in an office environment. Each sensor monitors the power consumption of any devices connected to the mains supply via the sensor, such as desktop computers. By monitoring power usage at the level of individual office users, building managers can better manage demand, detect potential faults in office equipment or identify environmental factors that affect power usage.

A major challenge for analysing this type of sensor data is how to help building managers extract potentially interesting usage patterns from the large volume of measurements generated by the sensor data streams. A drawback of fully unsupervised data mining technique in this context, such as cluster analysis, is their computational cost and the larger number of spurious patterns they may discover. Fully supervised data mining techniques are also of limited value in this context due to the lack of known events that can be used as labels for training examples. In this paper, we investigate the use of a semi supervised approach to power usage analysis, which is based on enabling users to query for usage profiles of interest.

A key research issue in this context is how to support usage queries that are expressive enough to represent a range of usage profiles, while still being computationally tractable. In this case study, we investigate the use of queries based on a predefined range of profile functions. When we encounter a data stream from a sensor node, we compute the similarity of the query

profile with the input stream. This similarity is reported as a membership function, and we study the spatial and temporal correlation among nodes with similar degree of membership to a given query profile.

We evaluate our proposed approach using simulated and real power monitoring data sets. The profiling of user behaviour is only one of the applications of our proposed method. Further, we detect regular (normal) as well as anomalous nodes, and report their mean and standard deviation of the membership values for each profile in the real power monitoring data. We also demonstrate that this methods has the ability to perform clustering on the simulated data set with high accuracy.

Profiling user behaviour in WSNs aids determining the strengths of various profile patterns in each node. Subsequently, they can be used for identifying normal and anomalous nodes in the network. In [1], [2], an algorithm for online detection and maintenance of motifs, which are repeated yet non-overlapping similar subsequences or patterns in a time series, is introduced. Another technique called Shapelets is introduced in [3], which are very small yet quite representative subsequences in a time series. They aid compression and classification of the time series data. Dynamic Time Warping (DTW) methods are introduced in [4] for time series comparisons. However, all these methods are computationally intensive in finding the patterns in the time series. In our work we present a query based technique which is computationally less complex, yet capable of detecting interesting patterns in the time series. After discovering such patterns, we can perform tasks such as clustering [5], classification [6] and anomaly detection [7], [8] in the monitored environment.

The contributions in this paper are two-fold. First, we present a method for fine scale analysis of user behaviour in power monitoring data. Second, this analysis can be performed efficiently and the results are quite interpretable. To the best of our knowledge, this is the first such fine scale analysis of power monitoring in an office environment.

The rest of the paper is organized as follows. Section II formulates the problem. Section III presents our proposed technique for power usage profiling. In Section IV, our proposed method is evaluated on a simulated and a real power monitoring datasets, followed by a discussion and conclusion in Section VI.

## II. PROBLEM STATEMENT

In a WSN, each sensor  $S_i$  generates a measurement sequence  $D_i$  which can be defined as a data stream. The sensor data  $D_i$  corresponds to a sequence of measurements  $x = \{x_t, t = 1, 2, \dots, n\}$  where  $n$  is the length of the stream. Each sample  $x_i$  is described as a  $d$  dimensional vector in  $\mathbb{R}^d$ . The elements of each vector are quantities measured by a sensor such as power consumption, light, vibration, temperature and movement. Therefore the input data in this application domain comprises the streams coming from the sensors  $S_1, S_2, \dots, S_k$  where  $k$  is the number of deployed nodes in the network. However, in this paper we are only dealing with one of the measurements, i.e., power consumption. Therefore  $x_i$  can be considered as a one dimensional vector or a single data point and the sequence  $x_1, x_2, \dots, x_n$  will be characterized as a time series. An example sequence is shown in Fig. 2(a).

In this work, we used a real-life power monitoring data set collected by the SmartCampus research testbed [9]. This data set comprises 250 programmable sensor nodes deployed over a three floor building. In this experimental setting, each desk is provided with an Internet of Things (IoT) node which observes 17 environmental features including energy consumption, which is measured in watts. Each node recorded various features at a sampling rate of approximately 10 seconds over an year. We considered a data set collected over a period of 2 weeks for our analysis.

*Profile and Expected Outcome:* We define a *profile* as a query that corresponds to a pattern associated with a specific behaviour in the network. In power monitoring sensor networks, a profile  $P$  can be demonstrated by the simple case of a step function:

$$P(t) = w * h * f(0, w; t) = h * (H(t) - H(t - w)) \quad (1)$$

According to equation 1,  $P$  returns the constant value  $h$  over the interval  $[0, w]$ , and zero for the rest of the values (see Fig. 2(b)).  $f(0, w; t)$  represents a uniform distribution over  $[0, w]$  and  $H$  corresponds to the step function:

$$H(t) = \begin{cases} 0 & : t < 0 \\ 1 & : t \geq 0 \end{cases} \quad (2)$$

The definition of such a profile stems from the power consumption trend in real data sets. An electricity usage sequence  $x_1, x_2, \dots, x_n$  often tends to demonstrate an abrupt increase or decrease in its magnitude. Afterwards it remains the same for a period of time  $T$  and then returns to the normal level. This is quite reasonable since energy usage is associated with users' behaviours during the day. The power consumption is monitored during the whole day but the user is present and consuming energy at specific periods of time. Therefore we state that an abrupt change has occurred at time  $j$  in a time series if the following constraint is satisfied.

$$\exists k_1, |x_{k_1} - x_j| \gg 0, \quad k_1 = j + \Delta, \quad \Delta \text{ is small.} \quad (3)$$

After the occurrence of  $l$  abrupt changes at times  $t \in \{a_1, a_2, \dots, a_l\}$ , i.e.,  $a_1$  is the time of the first abrupt change, the energy consumption level might remain the same for a period of time  $T_1, T_2, \dots, T_l$ , respectively. The value of  $T_i$  can decide which profile is the most appropriate match for the user behaviour. Therefore each profile is considered as

a query and our objective is to determine which profile is most representative of the observed user behaviour. A set of profiles  $P_1, P_2, \dots, P_h$  that have been chosen in the user query are applied on each time series  $[x_{1\dots n}]^j \in [x_{1\dots n}]^{1\dots k}$ . The outcome of this process is a sequence of membership degrees  $m_1, m_2, \dots, m_h$  for each profile (e.g., see Fig. 2(c)). It is worth noting that the values of  $T_i$  in the input time series and different magnitudes of  $w$  in our profiles are closely correlated. The membership function  $Q$  reflects the similarity of a user's behaviour to a specific query profile.

$$Q(P_i; x_{1\dots n}^j) = m_i, \quad 1 \leq j \leq k, \quad \forall j \quad (4)$$

*Assumptions and Success Metrics:* In order to evaluate the degree of membership between a time series  $x_1, x_2, \dots, x_n$  and a profile  $P_i$ , we assume that  $e$  and  $P$  are a pair of jointly wide-sense stationary stochastic processes. The procedure of determining such a membership degree for each pattern depends on the number of considered profiles,  $h$ . Therefore, in order to make the algorithm tractable, in this work, we do not consider complex queries and restrict the number of profiles to two. Moreover, since we are dealing with a real-life data set, we might encounter profiles at various times and locations. Therefore temporal shifts of profiles in the data are quite common. Thus, the proposed method must consider the possibility of temporal shifts in the data without the need for introducing new profiles.

Furthermore, one of the main challenges in data profiling is the dilemma of generality versus specificity. If the profiles are too tailored, they may not be able to present an overview of users behaving similarly. However if they are too general, they may consider all users to behave the same and do not give us individual insights with regards to the power usage behaviours of users.

## III. PROPOSED SCHEME FOR BEHAVIOUR PROFILING

In this section, we present the development of our scheme to profile the spatial and temporal behaviours of users in power monitoring data. The overview of the proposed method is depicted in Fig. 1. Since we are comparing the outputs of

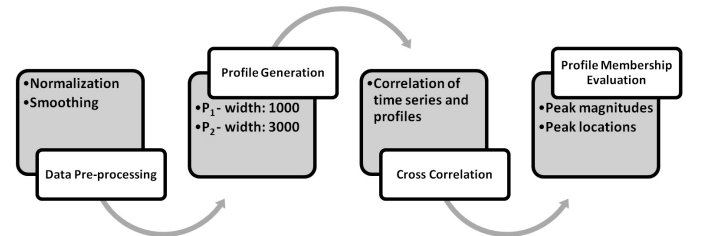


Fig. 1: Overview of the required steps in the proposed scheme.

sensors  $S_i$ , and our scheme needs to be indifferent towards variations in amplitudes, we have to normalize the outputs of each sensor as the first step in data pre-processing. In order to normalize the time series, we have employed z-normalization. After this process, each time series has zero mean and unit standard deviation. The normalized time series is denoted as  $x_1, x_2, \dots, x_n$ , where  $x_i = z\_norm(x_i) = \frac{x_i - \mu}{\sigma}$ ,  $\mu = \frac{1}{n} \sum_{i=1}^n x_i$  and  $\sigma^2 = \frac{1}{n} \sum_{i=1}^n x_i^2 - \mu^2$ .

The next step in data pre-processing is smoothing the normalized time series  $x'_1, x'_2, \dots, x'_n$ . Even though we have normalized the data, we still need to consider the possibility of noise in the time series. Therefore we utilize a moving average technique in order to smooth the possible noisy variations in our time series. The value of the parameter window,  $Win$ , for our time series is chosen to be  $Win = 72$  due to the nature of the input data. Since the time series records the daily behaviour of a user with a time interval of 10 seconds and we divide the daily activities of a user into three sections, we need to find a window which averages the data over an 8-hour period. The smoothed time series is denoted as  $x''_1, x''_2, \dots, x''_n$ , where  $x''_i = mov\_avg(x'_i, Win) = \frac{1}{2Win+1} \sum_{j=i-Win}^{i+Win} x'_j$ .

#### A. Profile Generation

The pre-processed input data  $x''_1, x''_2, \dots, x''_n$  is scanned once in order to determine the minimum and maximum magnitudes, i.e.,  $min_x, max_x$ . These values assist us in generating the profiles for detecting specific behavioural patterns. In order to reduce the computational overhead as well as increasing the interpretability of our scheme, we need to restrict the number of profiles. Therefore we consider two profiles  $P_1, P_2$  which have predefined periods  $w_1, w_2$  respectively..

The values of the periods  $w_1$  and  $w_2$  are determined empirically and according to the requirements imposed by the nature of the data that we consider. The values of  $w_1$  and  $w_2$  we used are 1000 and 3000 respectively. Therefore  $P_1$  records data profiles on a finer scale and  $P_2$  on a more general scale. Since there is no necessity for considering the reverse form of these profiles, we merely consider two profiles. In order to define the profiles  $P_1$  and  $P_2$ , we also need to adjust the value of parameter  $h$  which specifies the height of the change. We have to take the rise and fall patterns into account, thus we consider  $h$  to be  $max_x - min_x$ .

#### B. Cross Correlation

The next step after profile generation is to determine the similarity of the constructed profile to the original signal. The similarity between a pattern and signal can be determined through various means. Our selected metric for similarity evaluation is cross-correlation of the two signals. Cross-correlation is computed as a function of a time-lag that is applied to one of the input time series. This is appropriate for energy consumption time series since the patterns are shifted and various time lags enable us to detect these patterns in every location. This metric can be calculated according to equation 5 where  $f$  and  $g$  are the two input signals and  $f^*$  denotes the conjugate of  $f$ .

$$(f * g)[n] = \sum_{z=-\infty}^{\infty} f^*[z]g[n+z] \quad (5)$$

However, since we are considering finite time series, we can calculate the cross-correlation according to the following equation:

$$R_{fg}(z) = E[f_{n+z}g_n^*]; \quad z = 2n + 1 \quad (6)$$

We normalize the cross-correlation values in order to have an autocorrelation of 1 when none of the input signals have been shifted. As demonstrated in equation 6, the length of the

output signal is  $2n + 1$  where  $n$  is the length of the input time series.

#### C. Membership Degree Evaluation

The process of determining the membership degree of each signal with regards to a profile according to the cross-correlation output  $y_1, y_2, \dots, y_z$  consists of the following steps. Our initial objective is to determine the location  $l_i$  and magnitude  $v_i$  pairs of local maxima  $\langle l_1, v_1 \rangle, \langle l_2, v_2 \rangle, \dots, \langle l_p, v_p \rangle$  in the cross-correlation output signal. The distance among the maxima points must be no less than half of the pattern width,  $w$ . This means that we are considering 50% overlap. We consider the average magnitude of these peaks as well as the profile strength, i.e.,  $h$ , as an indication of the similarity between a pattern and signal. We refer to membership degree  $m$  as calculated in equation 7.

$$m = \frac{h}{p} \sum_{i=1}^p v_i, \quad v_i \geq \tau, \quad 1 \leq i \leq p, \forall i \quad (7)$$

The locations of the peaks can provide an insight when comparing the behaviour of individual time series. If a peak occurs at zero-lag, it is equivalent to the two signals being identical. However if maximum correlation occurs in other positions, we can determine the location of the pattern in the input signal as well. Therefore when comparing two signals' cross-correlation outputs  $y^1_{1\dots z}, y^2_{1\dots z}$ , we detect the highest peaks and compare their location  $l^1_i, l^2_j$  against each other. We calculate the distance among these locations and decide whether the most significant patterns of these two signals occurred in each other's vicinity.

$$|l^1_i - l^2_j| \leq \tau_d; \quad v^1_i \geq v^1_k, \quad v^2_j \geq v^2_k, \quad 1 \leq k \leq p, \quad \forall k \quad (8)$$

We demonstrate the problem through an illustrative example depicted in Fig. 2. In the following example, the input signal,  $x_1, x_2, \dots, x_n$ , come from sensor  $S_1$  and the query profile is denoted as  $P$ . The cross-correlation output is depicted in the bottom figure and the red triangles represent the  $\langle l_i, v_i \rangle$  pairs. The membership degree of such a profile for the input signal is calculated as the average of values  $v_1$  and  $v_2$ . The locations,  $l_1$  and  $l_2$  correspond to the positions of abrupt changes in the input signal,  $a_1$  and  $a_2$  respectively. The pseudocode of

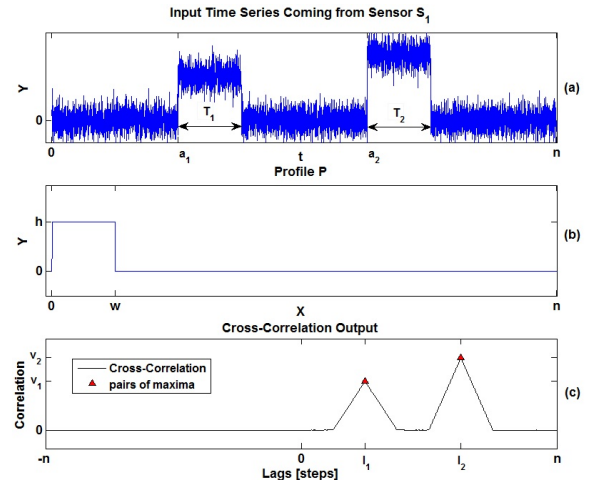


Fig. 2: An example of determining the membership degree of a profile.

the algorithm for computing the membership degree of profiles with an input signal is depicted below. However, it is worth noting that due to the definition of the profiles, i.e., a simple step function, we have less computational complexity than the following pseudocode for determining cross-correlation. The computational complexity of the pseudocode is  $O(n^2)$  while in our technique the same task can be performed in  $O(n)$ . Since the width of the step is fixed in each profile and remains zero for the rest of the values, the cross-correlation can be computed in a linear complexity of the input time series length  $n$ .

---

**Algorithm 1:** ProfileMembership

---

**Data:**  $x = (x_1, x_2, \dots, x_n)$ ,  $x$  is the input signal.

**Result:**  $m = (m_1, m_2)$ ,  $m$  is the membership degree of our Profiles.

---

```

1 begin
2    $w1 \leftarrow 1000, w2 \leftarrow 3000$  ; // Profile Widths
3    $win \leftarrow 72$  ; // Smoothing Window
4    $x_n = \text{normalize}(x)$  ; // Normalized Input
5    $x - n_s = \text{smooth}(x_{n_s})$  ; // Smoothed Input
6    $h \leftarrow \max(x_{n_s}) - \min(x_{n_s})$  ; // Profile Height
7    $p_1(t) = h(H(t) - H(t - w1))$  ; // Profile  $p_1$ 
8    $p_2(t) = h(H(t) - H(t - w2))$  ; // Profile  $p_2$ 
9    $\mu_{p_1} \leftarrow \frac{h*w1}{n}, \mu_{p_2} \leftarrow \frac{h*w2}{n}$  ; // Profile Mean
10   $s_x \leftarrow 0, s_{p_1} \leftarrow 0, s_{p_2} \leftarrow 0$  ; // Profile Sigma
    /* Cross-Correlation Parameters Computation */
11  for  $j \leftarrow 1$  to  $n$  do
12     $s_x += (x_{n_s}(j) - \mu_{x_{n_s}})^2$ ;
13     $s_{p_1} += (p_1(j) - \mu_{p_1})^2$ ;
14     $s_{p_2} += (p_2(j) - \mu_{p_2})^2$ ;
    /* Cross-Correlation Computation */
15  for  $delay \leftarrow -n$  to  $n$  do
16     $s_1 \leftarrow 0, s_2 \leftarrow 0$ ;
17    for  $i \leftarrow 1$  to  $n$  do
18       $j \leftarrow i + delay$ ;
19       $s_1 += (x_{n_s}(i) - \mu_{x_{n_s}}) * (p_1(j) - \mu_{p_1})$ ;
20       $s_2 += (x_{n_s}(i) - \mu_{x_{n_s}}) * (p_2(j) - \mu_{p_2})$ ;
21     $y_1(delay) = s_1 / \sqrt{s_x * s_{p_1}}$ ;
22     $y_2(delay) = s_2 / \sqrt{s_x * s_{p_2}}$ ;
23   $peaks_1 = \text{detect\_peaks}(y_1)$ ;
24   $peaks_2 = \text{detect\_peaks}(y_2)$ ;
25   $m_1 \leftarrow h * \mu_{peaks_1}$ ;
26   $m_2 \leftarrow h * \mu_{peaks_2}$ ;
27 end
```

---

#### IV. EXPERIMENTAL RESULTS

In this section, we demonstrate the ability of the proposed algorithm for profiling user behaviour in both simulated and real data sets through an empirical study. We also consider anomaly detection and clustering outcomes in the real data set.

*1) Simulated Dataset:* In order to evaluate the capability of our algorithm, we created a synthetic test-bed in order to simulate known data patterns. The patterns that we have considered for data simulation consist of the common rise and fall structures and a possible combination of them. We describe

these patterns in the following equations:

$$P_1(t) = h_1 * [H(t) - H(t - w_1)] \quad (9)$$

$$P_2(t) = -h_2 * [H(t) - H(t + w_2)] \quad (10)$$

$$P_3(t) = h_3 * [H(t + w_3) - H(t - w_3)] \quad (11)$$

These patterns are modified by increasing or decreasing the height  $h_i$  and width  $w_i$  in order to study their effects on the cross-correlation outcome. However the length of the profile is always increased through zero-padding in order to calculate its cross-correlation with a longer time series.

The simulated data set consists of 13 nodes (signals) in which there are always five time series with rise patterns  $P_1(t)$ , and four time series which are constructed from a fall pattern  $P_2(t)$  and the rest are based on a rise-fall pattern  $P_3(t)$ . The length of the time series is 8000 observations. We considered various scenarios such as doubling or halving the width of the pattern as well as injecting several instances of these patterns in various positions of the time series. In the next section, we depict some results obtained by evaluating the cross-correlation among constructed times series and patterns. We have also added Gaussian white noise in order to determine the tolerance of the cross-correlation and correlation coefficient to noisy data sets. The signal to noise ratio was set to 10 in our simulation.

In our simulation we divided the 13 signals (nodes), mentioned above, into 3 *clusters*. The members of each cluster are constructed based on only one of the three defined patterns, i.e.,  $P_1, P_2, P_3$ . Sample cluster members are depicted in Fig 3(a).

Throughout the simulation, we compared each signal against all three basic patterns. However according to the results, we realized that only one of the patterns is sufficient for analysing the signal's shape. The result of cross-correlation of signals with one basic pattern is depicted in Fig 4(a). As you can see, the signals are only compared against one of the profiles and the outcomes are quite representative of signals' behaviours. Evaluating the cross-correlation between a simple profile and time series can reveal the existence, location as well as the strength of the pattern. The location and height of the positive and negative peaks correspond to the time-interval and support of the profile that characterizes the behaviour of a node. As depicted in Fig 4(a), if we detect a positive peak in the cross-correlation outcome in the negative lag, our signal is based on  $P_2$ . However if we detect negative and positive peaks in the positive lag, our signal is based upon  $P_3$ . The last case which is not depicted in Fig 4(a), is when a signal is based on  $P_1$ . In such a case, the cross-correlation outcome demonstrates positive peaks in the positive lag. This information can be used for clustering the simulated signals.

*2) Real Dataset:* We have evaluated our proposed method on a real power monitoring dataset. This data is collected by 256 programmable sensor nodes which are deployed in various offices of a three story building [9]. This experiment is the basis of the Smart Campus facility. The nodes capture 17 real valued features. We focus on the measurements that each sensor makes of the electric power consumption made by a user at that office location.

In this study, we have considered the power usage time series in order to study the possibility of clustering the nodes

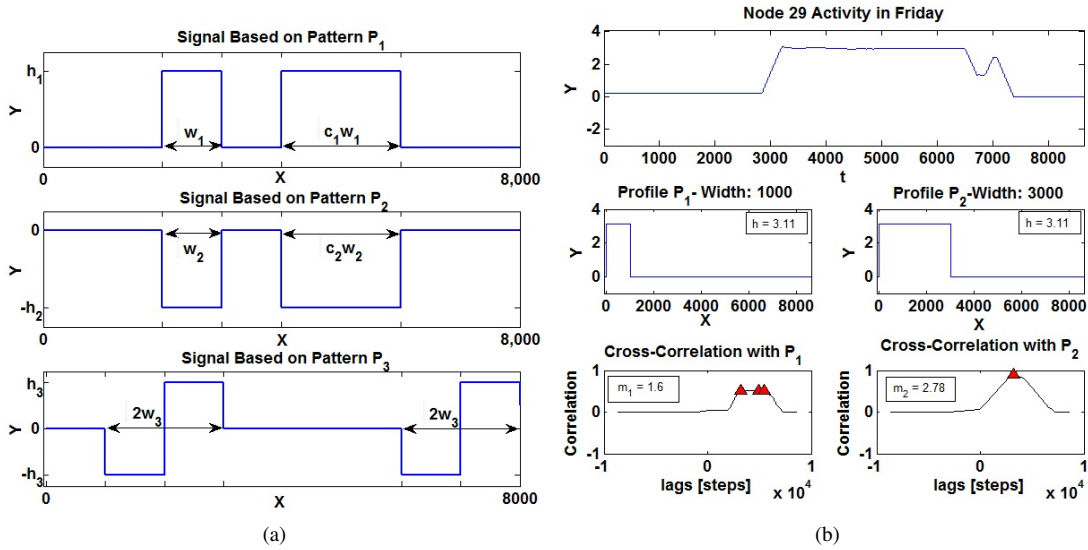


Fig. 3: (a) A sample of three possible signals in our simulation dataset where  $c_1$  and  $c_2$  are randomly chosen constant values. (b) Node 29 activity during Friday, this membership degree remains the same during other active days as well.

as well as for anomaly detection and behaviour profiling. The dataset consists of recordings over a period of two weeks sampled approximately every 10 seconds. We were also provided with temporal and spatial information, i.e., the date and the office number of each node in the WSN. A thorough description of this dataset can be found in [9]. In this study we used data from 120 nodes distributed over three levels of the building, recorded between 1/4/2012 and 15/4/2012.

#### A. Profiling User Behaviour

As mentioned earlier, we considered two profiles to characterise the power consumption trends in the real data. we used two different widths to record both fine and coarse scale information. The first profile is using the *width* = 1000, and the second profile is using the *width* = 3000. Another condition that needs to be satisfied in order to consider a time series for profiling is the magnitude of its variation. The minimum and maximum value of a signal must exceed a threshold of 1 in order to consider the time series as an interesting or *active signal*, i.e.,  $\max(\text{signal}) - \min(\text{signal}) > 1$ . Therefore flat signals or time series with very low fluctuation levels are considered as *inactive* time series.

In order to determine whether a user is exhibiting consistent behaviour, we need to calculate the membership values of each node for every day of the week. If the node exhibits similar membership values to the profiles, we can conclude that the user demonstrates regular (normal) behaviour. For instance Node 29 deployed in Office 32, first floor, demonstrates such consistent behaviour during the active days of the week. The nodes' behaviour and the membership degree outcomes are depicted in Fig. 3(b). The computed means and standard deviations of node 29's membership to profiles  $P_1$  and  $P_2$  during its *active period*, i.e., the days of the week where the fluctuations of the power consumption is significant enough to be considered active (i.e.,  $\max(\text{signal}) - \min(\text{signal}) > 1$ ), are  $1.57 \pm 0.06$  and  $2.70 \pm 0.1$  respectively. A large value for the mean reveals how strong the pattern is in the signal, and the small standard deviation value reveals that the signal is consistently showing such membership during other

days as well. Table 5(a) provides the outcome of the defined membership function for 24 nodes (out of the 120 nodes) in the building. The rest of the nodes results are omitted due to page limitations. These analysis help identifying users with consistent behavioural patterns as well as irregular users in the office. Furthermore, we can determine users' interactions in terms of their power usage profile with each other in an office. Therefore individual nodes can be investigated on a fine scale in order to provide insight into user behaviours.

#### B. Clustering Simulated Dataset

In the beginning phases of this study, we considered three different patterns as the profiles defined in Equation 9. The simulated data was generated such that the data would be clustered into three groups. The membership degrees calculated in this section are different from what we have described so far. In order to cluster the data, we divide the cross-correlation outcome into two sections: negative and positive lags. Moreover, we detect valleys and peaks in the cross-correlation signal. Every membership outcome is composed of the following sections: average value of positive and negative peaks in negative as well as positive lags which are denoted as  $\{p\_nlag, n\_nlag, p\_plag, n\_plag\}$  respectively.

The signals based on pattern  $P_1$  demonstrate peaks in positive lags. On the other hand, signals based on pattern  $P_2$  result in peaks in negative lags. Finally the signals based on pattern  $P_3$ , lead to both negative and positive peaks in positive lags. Through clustering, we assign instances to clusters 1, 2 and 3 based on  $\{p\_nlag, n\_nlag, p\_plag, n\_plag\}$ . Our proposed method was able to successfully determine the patterns belonging to each cluster. We also added Gaussian white noise to the input data, and cross-correlation was still able to assign acceptable membership degrees to each input signal. During this study, we observed that there was no need for complex patterns since these patterns can be constructed by simple yet representative patterns. In the discussion section, we demonstrate how a simple pattern corresponds to more complex or modified profiles. Fig. 4(a) depicts the outcome of clustering for three sample signals. The figure in the top



is showing a signal coming from profile  $P_2$  and the bottom figure is showing the signal coming from profile  $P_3$ . Here we demonstrate that the membership values for the signals coming from different clusters are different from each other. These two signals are representatives of their clusters. It can be noted that we achieved 100% accuracy and 0 false positives on this simulated data set.

### C. Anomaly Detection

Another application of profiling user behaviour is anomaly detection in an office environment. Anomalies are defined as instances that do not conform to the previously observed normal behaviour. One of the major concerns to be taken into account in our study is the period of time when we monitor power consumption. This period of time consists of weekends and weekdays, thus we do not expect to observe *strong* profiles during every day of the week. This is the basis of anomaly detection in behaviour profiling.

In the context of the data that we are studying, we define the anomalies as those instances which exhibit significant power consumption fluctuations during *weekends*. The normal behavior is defined based on previously seen data. For instance, if we know that the nodes cannot show activity during weekends, and the nodes that we have seen so far are almost flat lines (inactive) during weekends, then the normal behavior can be interpreted as those nodes that always show significant fluctuations only during weekdays.

According to our definition of normal behaviour, we can state that the node must show significant membership value during the weekday and very low membership value during the weekend. The anomalous nodes are demonstrating significant membership values during weekend. Therefore they are exhibiting considerable power consumption fluctuations independent of the days. However, if they were normal nodes, they would show strong membership values only during weekdays. The anomalous nodes do not need to be consistently showing the same behavior during the weekends. The fact that they are demonstrating significant fluctuation is sufficient for us to say that they are irregular.

The anomalous nodes require further investigation and may also help us to decrease energy usage levels in the building. For instance, two nodes, 108, 104, in office 22, second floor, demonstrate power consumption during all days of the week and even in some cases, the power usage was more severe during weekends. They have the different membership degrees yet they all show some existing profiles without any dependence upon the day of the week, i.e., weekends and weekdays. The user profiles are demonstrated in Fig. 4(c) during a weekend. The arrows in Fig. 4(b) point to the locations of the anomalous nodes as well as their corresponding office. Some other anomalous nodes are shown in red in Table 5(a).

## V. DISCUSSION

The results in the previous section demonstrate the application of user behaviour profiling in power monitoring tasks. Although we have used three basic patterns as profiles in simulated data, through our experiments, we realized that a single simple pattern can be representative of the time series. The membership function can be modified in order to help us

identify various time series. For instance, as depicted in the clustering section, we can detect valleys and peaks and also consider negative and positive lags in order to determine the patterns with which we are dealing.

A potential issue for this membership function occurs when we have multiple instances of a pattern in our signal. Since the mean value can be sensitive to the largest and smallest value in a set, we may end up with a lower or higher membership degree in a signal. An example of this scenario can be seen in Fig. 5(b). This can be addressed by either redefining the membership function or considering another profile in which we have multiple occurrences of a single pattern.

## VI. CONCLUSION AND FUTURE WORK

In this paper, we have presented a preliminary study on profiling user behaviour in power monitoring data. We have demonstrated that we can detect anomalous nodes by computing a membership function based on cross-correlation. The profiles provide a means for analysing user behaviour at a fine scale. Moreover, since the pre-defined patterns are simple, we can perform high level data mining tasks such as clustering and anomaly detection with reasonable computational complexity. We have shown the capability of our scheme through evaluation on synthetic as well as real data sets. The research is still on-going, and there are several directions for future work. More complex patterns can be defined as a basis of comparison in order to classify nodes in more detail. We can consider other functions instead of the step function to build new patterns. Moreover, the detection of patterns can be done automatically and in an online manner.

## ACKNOWLEDGMENT

We thank the support from the NICTA; the REDUCE project grant (EP/I000232/1) under the Digital Economy Programme run by Research Councils UK - a cross council initiative led by EPSRC and contributed to by AHRC, ESRC and MRC; the EU FP7 SocIoTal grant and the Australian Research Council grants LP120100529 and LE120100129.

## REFERENCES

- [1] A. Mueen and E. Keogh, "Online discovery and maintenance of time series motifs," in *KDD*, 2010, pp. 1089–1098.
- [2] H. T. Lam, T. Calders, and N. Pham, "Online discovery of top-k similar motifs in time series data," in *SDM*, 2011, pp. 1004–1015.
- [3] A. Mueen, E. Keogh, and N. Young, "Logical-shapelets: An expressive primitive for time series classification," in *KDD*, 2011, pp. 1154–1162.
- [4] T. Rakthanmanon, B. Campana, A. Mueen, G. Batista, B. Westover, Q. Zhu, J. Zakaria, and E. Keogh, "Searching and mining trillions of time series subsequences under dynamic time warping," in *KDD*, 2012, pp. 262–270.
- [5] M. Moshtaghi, C. Leckie, S. Karunasekera, J. C. Bezdek, S. Rajasegarar, and M. Palaniswami, "Incremental elliptical boundary estimation for anomaly detection in wireless sensor networks," in *IEEE ICDM*, 2011, pp. 467–476.
- [6] S. Rajasegarar, C. Leckie, J. C. Bezdek, and M. Palaniswami, "Centered hyperspherical and hyperellipsoidal one-class support vector machines for anomaly detection in sensor networks," *IEEE Trans. on Info. Foren. and Sec.*, vol. 5, no. 3, pp. 518–533, Sept 2010.
- [7] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 15:1–15:58, July 2009.

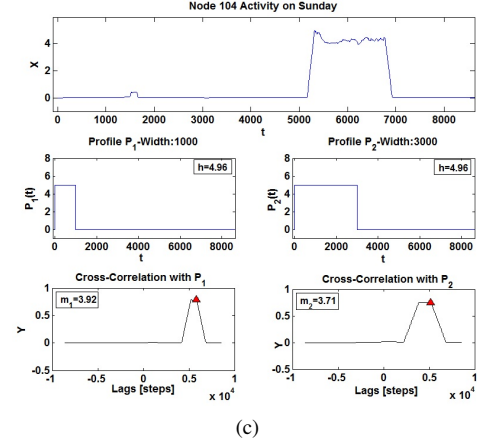
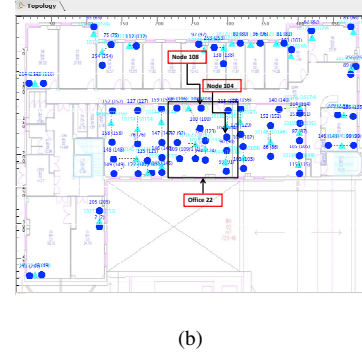
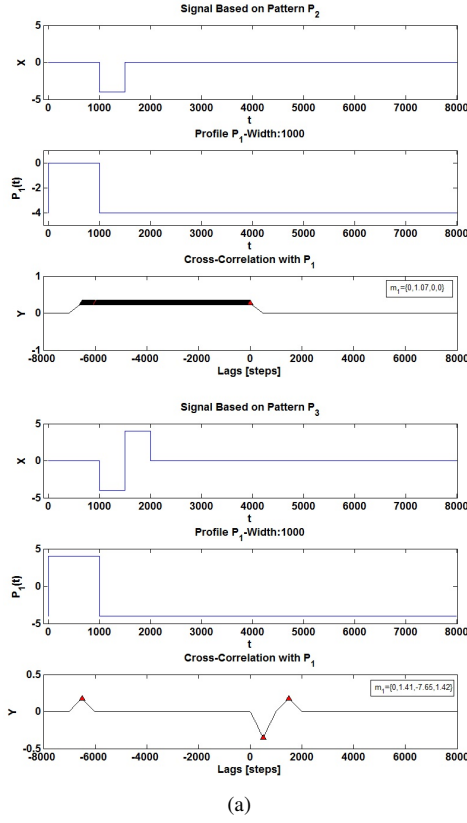


Fig. 4: (a) Two clusters and their corresponding outcomes. (b) Node and gateways deployment of second floor, anomalous nodes, 108, 104, and their corresponding office, 22. (c) Node 104 activity during Sunday, this membership degree demonstrates significant power consumption during weekend.

Node ID	Active Period	Weekend	Weekday	Node ID	Active Period	Weekend	Weekday
	$\mu_{P_1} \pm \sigma_{P_1}$ $\mu_{P_2} \pm \sigma_{P_2}$	$\mu_{P_1} \pm \sigma_{P_1}$ $\mu_{P_2} \pm \sigma_{P_2}$	$\mu_{P_1} \pm \sigma_{P_1}$ $\mu_{P_2} \pm \sigma_{P_2}$		$\mu_{P_1} \pm \sigma_{P_1}$ $\mu_{P_2} \pm \sigma_{P_2}$	$\mu_{P_1} \pm \sigma_{P_1}$ $\mu_{P_2} \pm \sigma_{P_2}$	$\mu_{P_1} \pm \sigma_{P_1}$ $\mu_{P_2} \pm \sigma_{P_2}$
13	2.21±0.74 2.50±0.86	1.14±1.36 0.96±0.97	1.30±1.32 1.52±1.62	64	4.40±0.46 4.80±0.70	0.00±0.00 0.00±0.00	1.32±2.14 1.44±2.34
17	1.57±0.53 2.53±0.92	0.00±0.00 0.00±0.00	0.78±0.00 1.27±1.47	97	1.20±0.46 1.70±0.48	1.30±0.00 1.39±0.88	1.21±0.35 1.75±0.47
18	2.50±0.92 3.40±0.84	0.22±0.49 0.00±0.00	1.30±1.50 1.70±1.88	102	1.80±0.69 2.18±0.70	1.54±1.12 2.04±1.49	1.57±0.84 1.81±0.75
19	2.75±0.87 4.61±1.68	0.00±0.00 0.00±0.00	0.82±1.39 1.38±2.36	104	4.00±0.59 3.35±0.25	5.30±1.88 2.76±1.55	0.40±1.27 0.30±0.94
21	2.70±1.41 2.22±1.10	1.82±1.21 1.57±1.32	1.52±2.00 1.21±1.49	105	1.40±0.38 1.52±0.63	1.08±0.78 0.71±0.46	1.29±0.53 1.62±0.76
26	2.09±0.13 3.38±0.21	0.00±0.00 0.00±0.00	1.04±1.10 1.69±1.79	108	1.30±0.27 1.71±0.61	1.43±0.27 1.88±0.57	1.11±0.46 1.45±0.79
29	1.57±0.66 2.70±0.10	0.00±0.00 0.00±0.00	0.78±0.63 1.35±1.43	112	5.34±0.90 4.07±1.22	0.00±0.00 0.00±0.00	1.62±1.81 2.03±2.29
52	2.14±0.05 3.58±0.16	0.00±0.00 0.00±0.00	0.86±1.11 1.43±1.85	114	1.94±0.54 2.25±0.69	1.95±0.37 1.98±0.72	1.94±0.63 2.38±0.67
53	1.75±1.01 2.43±1.22	1.56±1.41 1.98±1.47	1.30±1.66 1.92±1.55	125	1.80±0.65 2.29±0.91	1.15±1.19 1.05±1.15	1.70±0.65 2.45±0.89
54	1.50±0.39 2.95±0.50	0.00±0.00 0.00±0.00	0.95±1.03 1.48±1.59	127	0.86±0.19 1.06±0.17	0.80±0.09 1.04±0.11	0.89±0.22 1.07±0.19
55	1.32±0.13 2.26±0.23	1.38±0.07 2.49±0.11	1.28±0.15 2.20±0.25	138	1.27±0.78 1.54±0.69	0.89±0.54 1.12±0.69	1.08±1.01 1.29±0.99
60	1.59±0.13 2.73±0.18	0.00±0.00 0.00±0.00	0.80±0.84 1.36±1.44	145	2.03±0.69 2.25±0.59	1.71±1.01 1.47±0.98	1.58±1.11 1.97±1.12

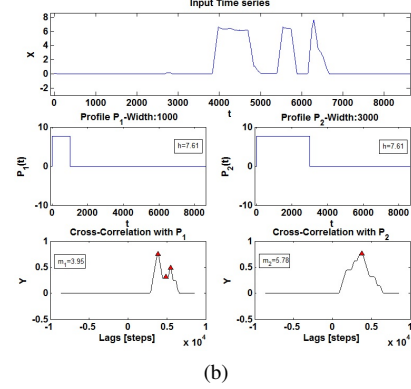


Fig. 5: (a) Outcome of membership function in terms of mean and standard deviation for weekend, weekday and active period. (b) An example scenario of the issue with taking the mean value as the membership degree.

[8] S. Rajasegarar, C. Leckie, and M. Palaniswami, "Anomaly detection in wireless sensor networks," *IEEE Wireless Comms.*, vol. 15, no. 4, pp. 34–40, 2008.

[9] M. Nati, A. Gluhak, H. Abangar, and W. Headley, "Smartcampus: A user-centric testbed for Internet of Things experimentation," in *IEEE WPMC*, June 2013.